**Day 8 Task: Basic Git & GitHub for DevOps Engineers.**

**#owner : vishal saxena**

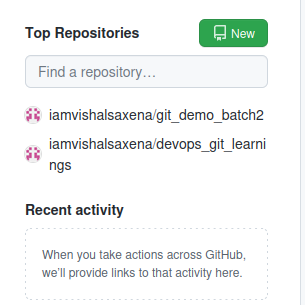
**Exercises:**

**Q. Create a new repository on GitHub and clone it to your local machine**

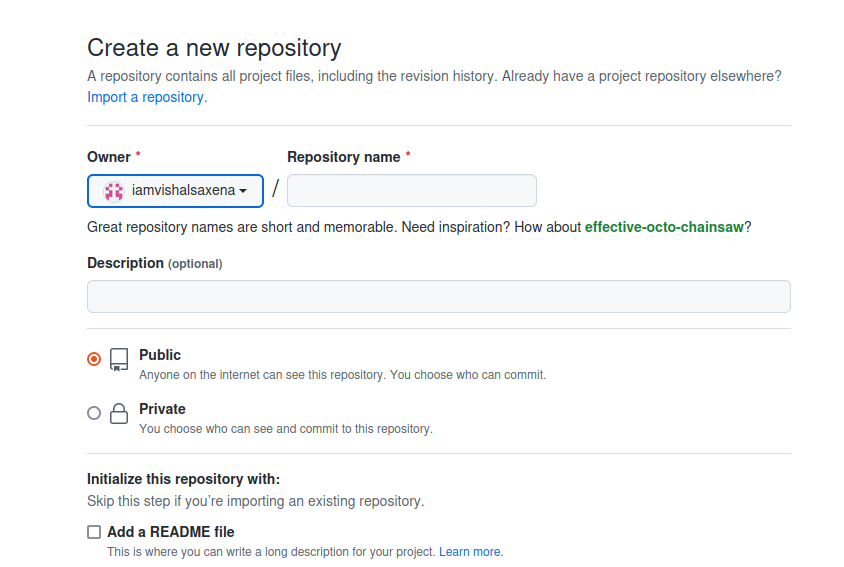
We can create a new repository by following the below steps if you are creating a new repository for the very first time.

Step 1. Click on the green button which says “NEW” next to the Top Repositories section on the left of your Git dashboard page.

See the below image for reference as I am talking about the green button below.

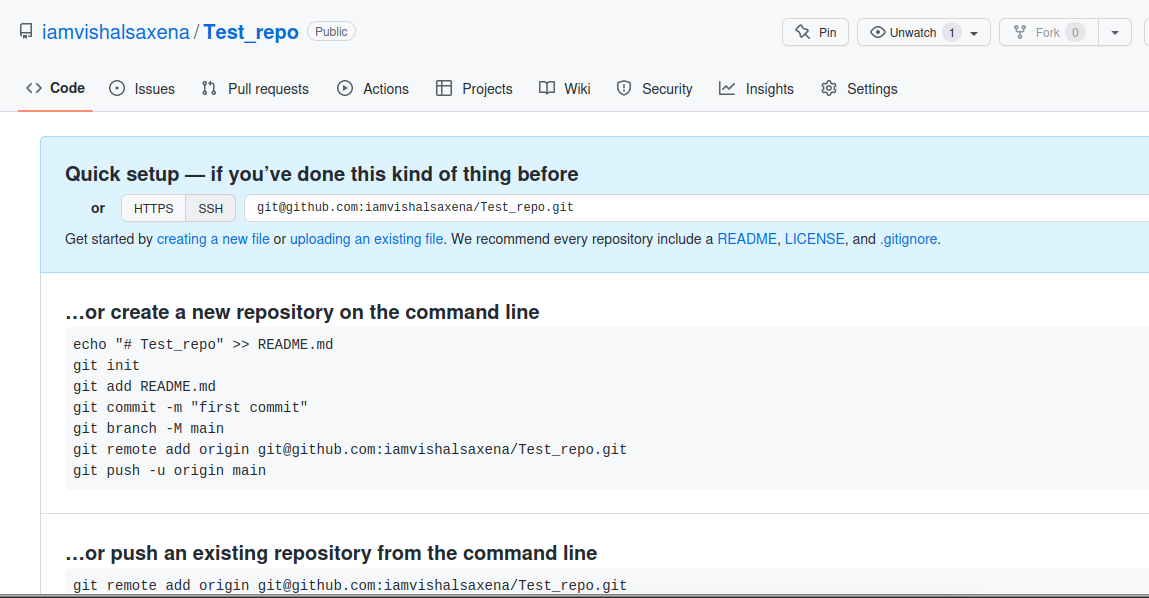


Step 2. By clicking the green button the button redirects you to page like below : -



Just fill out all the details like repository name and select the button to put this repo either public or private and next select the README file for the sake of your flexibility and but for now I am not selecting the README file.

which gives a page something like below.



It’s Done you have successfully created your new repository by just following few steps.

Now , the hard part starts which includes interaction of terminal of your local system.

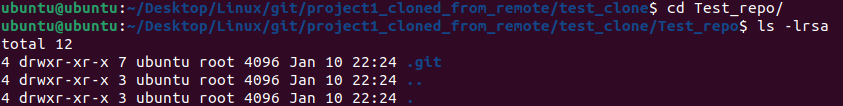
Don’t get stressed out , chill just kidding its easy job to do , just follow my steps and guide.

For Cloning the same on your local system there is two ways either you can opt the HTTPS way or you can choose the slight hard way which would be by using SSH.

So, let’s do it using HTTPS way first : -

1. Create a folder where you want to clone your newly created repository and open up your terminal in the same folder.
2. Next , type **git clone <your repo url>**  which you want to clone .  
     
   eg – Here I am cloning my newly created repo in my newly created folder “**test\_clone folder”.** Refer the below screen grab for better understandings.  
     
   Note: - Ignore warnings as my repo doesn’t have any file in it .  
    

So , hopefully you will get the .git hidden file in your local folder which looks something like this:-



Cheers guys !! it’s done without breaking a sweat .

Now for the SSH part , if you want to skip this section its totally fine most people skip the hard way :) as through this way we are trying to achieve the same goal to clone the repo.

Note : - **By doing this you don’t have to enter your password again and again when performing push operation as the public key is set on the remote and the private key is stored in your local which create more secured connection between your remote repository (GIT,GITHUB or BITBUCKET) and your local repository (which is on your local system folders) .**

So, let’s do it using SSH way : -

Step 1. We need to first create and setup the SSH Keys on your remote as well as in your local system.

Use the following command to generate your own ssh keys

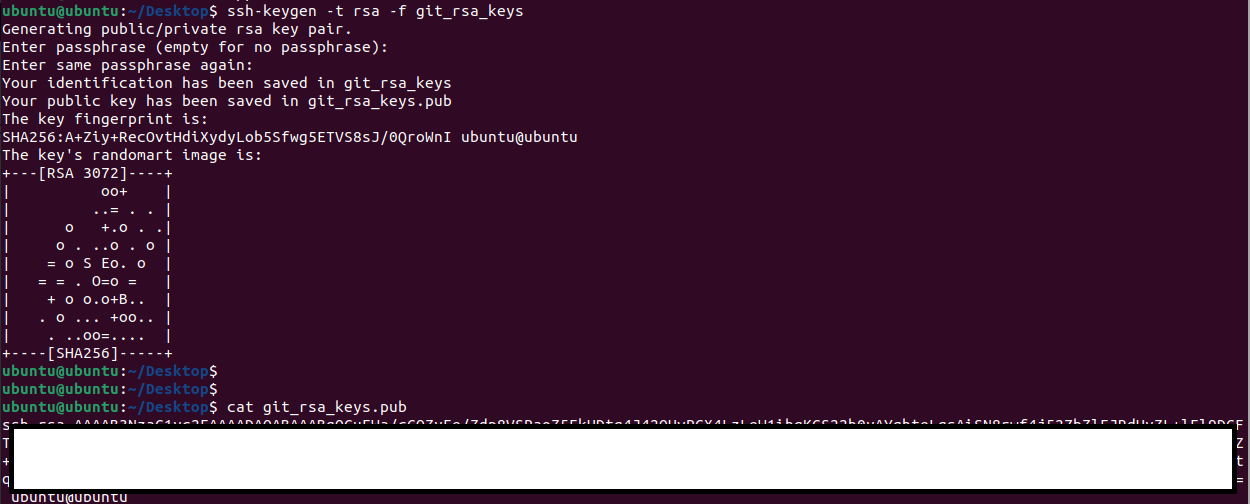
**ssh-keygen – t rsa -f git\_rsa\_keys**

ssh-keygen --> it is command to generate the ssh keys

-t rsa --> Denotes the type of encryption ssh keys you want to generate

-f git\_rsa\_keys --> Denotes the file name of the keys

By running the above command gives you .pub and normal private key file in which the content of the .pub file is going to paste on github online.



**Its better if you put your keys in the below path which and generate the keys on the same path.**

**Go to your ~/.ssh folder To generate new keys ssh-keygen -t rsa.**

**And Edit the config file and paste the following data for storing your public key in your .ssh folder. {In case you can’t able to locate the config file in .ssh folder(hidden folder) create your own config file and paste the following.}**

Host github.com   
 AddKeysToAgent yes   
 UseKeychain yes   
 IdentityFile ~/.ssh/git\_rsa\_keys  
  
**Remove the UseKeychain line if you are not on your mac OS.**

So we are good with our local setup of SSH Keys now put your public key on your GITHUB account .

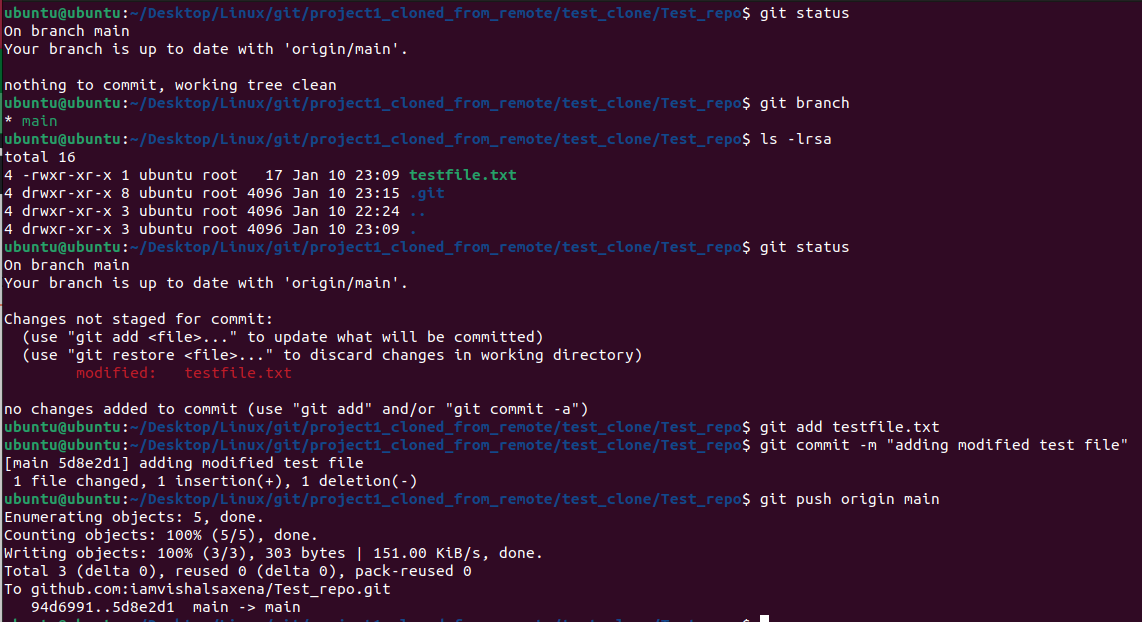
Go to your setting -> access --> SSH and GPG Keys --> and click on NEW SSH KEY to add you pub key (just open git\_rsa\_keys.pub and paste the content over here. )

Now just do the same process type **git clone <your repo url>** which you want to clone .

And All Done !!!

**Q. Now , Let’s make some changes to a file in the repository and commit them to the repository using Git.**

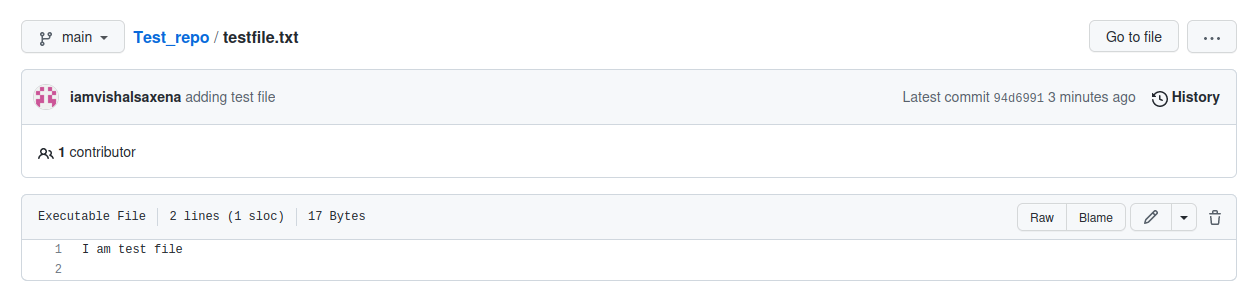
Follow the command to add the modified file then commit it and lastly push it remote repository on github  
   
﻿**git status  
﻿(The above command will give the current status of your local repository)  
  
git add testfile.txt  
﻿(The above command will add the file to the staging area if the file showed modified in git status)  
  
git commit -m "adding modified test file"  
﻿(The above command will commit the file and move the file from staging area, commit is an operation that records changes to a repository. When you commit changes, you are creating a new "snapshot" of the repository that includes all of the changes you've made up to that point )  
  
git push origin main  
(The above command pushed all your changes to your remote repository)**

****

**Q. Push the changes back to the repository on GitHub**

**The following command helps us to push the changes to remote repository   
(Please refer the previous page for more details.)  
  
git push origin main  
(The above command pushed all your changes to your remote repository)**

testfile.txt before modification on remote (Github)



After doing modification in testfile.txt

